Cooperative Multi-Robot Navigation and Mapping of Unknown Terrain

Adrian Boeing, Thomas Bräunl, Robert Reid, Aidan Morgan, Kevin Vinsen

***Abstract*— We describe the hardware, software and sensor equipment for a fleet of mobile robots used for cooperative multi-robot navigation and mapping. This robot team was developed for the MAGIC 2010 Robotics Challenge and successfully made it through two down-selection processes to the finals, where it placed 4th. Our mobile robot design leverages off-the-shelf hardware and open source software with novel software contributions to realize cooperative navigation, mapping, reconnaissance, and surveillance tasks for a large urban environment.**

I. Introduction

THE Multi Autonomous Ground-robotic International Challenge (MAGIC) 2010 continues the history of defense funded robotics competitions, including the DARPA Grand Challenge 2004/2005 [1] and the DARPA Urban Challenge 2007 [2]. The challenge was organized by the U.S. RDECOM and the Australian DSTO. From a field of international applicants, twelve teams were selected for the preliminary round, where each team had to complete a number of navigation/mapping tasks. From these, six finalists were invited to compete in the robotics challenge at the Adelaide Showgrounds in South Australia.

MAGIC 2010 was a traditional "system-of-systems" integration challenge. Teams had to produce vehicles that successfully integrated hardware (sensors, vehicle platform) and software (sensing, planning, mapping, team coordination) into a cohesive unit capable of meeting the challenge's requirements. This included implementing communication and coordination strategies for a group of vehicles to work collaboratively for a number of tasks. In particular, MAGIC focused on minimizing the human interaction in controlling a fleet of heterogeneous ground robots.

Our entry, team MAGICian, comprised seven WAMbot robots (Western Australia MAGIC Robot), based on a Pioneer 3AT outdoor robot platform, an automotive core-2 duo PC under Windows XP, three laser scanners (Sick LMS, Sick Ibeo and Hokuyo), a Qstarz GPS, an Xsens IMU, and two digital cameras for teleoperation and object identification/tracking.

Manuscript received March 31, 2011. This work was supported in part by the Air Force Research Laboratory, under agreement number FA2386-10-4024U.S. as well as through a grant by Thales Australia.

A. Boeing, T. Bräunl, R. Reid, A. Morgan, K. Vinsen are with The University of Western Australia, Perth (e-mail: thomas.braunl@ ieee.org).

A. Boeing is also with Edith Cowan University, Perth.

Fig. 1. WAMbot robot team

II. Challenge Tasks and Solution Strategy

A. Tasks

The challenge comprised of three stages with increasing levels of difficulty. In each phase, the robot team had to autonomously explore and map a complex outdoor/indoor environment with minimal operator intervention. Any operator intervention in excess of ten minutes resulted in points being deducted. Robot teams had to explore the environment automatically, teleoperating or waypointing a robot was not allowed (except for moving a robot out of a stuck position, while penalties applied). All robots were networked and needed to maintain wireless communication in order to operate for safety reasons. Stationary as well as moving "Objects of Interest" (OOI) had to be detected and entered into the map, while "hostile" OOIs had to be identified by using an on-board laser pointer. All OOIs were color-coded to help with recognition.

B. Solution Strategies

A system-of-systems architecture was used for the WAMbot robot team, including several layers of redundancy.

Dependability / Fault Tolerance: Our solution is designed to be redundant at several levels. We use seven robots, so the loss of a robot can be compensated for by the rest of the team. In this case the robot network automatically reconfigures itself. Mission control (and point scoring) depends on multiple, redundant ground control stations, which are connected to all robots via a wireless link. As this is a crucial component, we are using two identical ground control stations (GCS) in a redundant configuration (see Figure 3). Finally, we chose

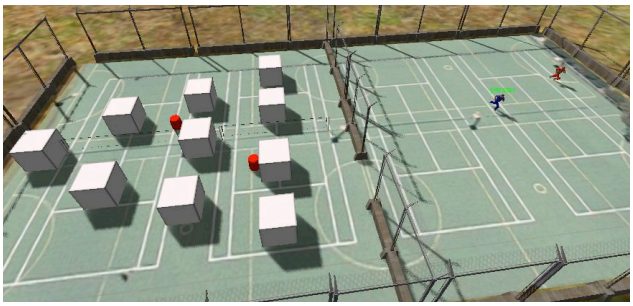

Fig. 2 Preliminary round navigation task in simulation environment (image courtesy of Martin Masek, ECU)

Digital Distribution Service (DDS) [3] as a middleware for inter- as well as intra-robot (and GCS) communication. DDS includes fault-tolerant features such as restarting of individual failed processes.

Sensor fusion: Integrating odometry, IMU, GPS, laser scanners and cameras for improved accuracy in navigation and mapping

Multi-robot mapping: A novel distributed multi-agent simultaneous localization and mapping (SLAM) approach provides robots and the GCS with a global map of the environment, and provides accurate global pose estimates for each robot within the map.

Object recognition: Since all OOIs are color coded, a visual color histogram-based approach was used in combination with laser range finder data.

Simulation: In simulation trials before the challenge, testers assumed the role of either an autonomous vehicle, mobile adversary, or non-combatant. An overhead view of the area was provided from a separate camera so that scenarios could be observed (see Figure 2). Each team member participated from their own computer over a network, so there was no need for all participants to be located in the same physical area

Testing: "Use-Cases", describing the system's behavior as it responds to operator requests were used in combination with extensive test plans and procedures in order to debug the system and prevent failures that could have resulted in the potential loss of a robot.

III. Multi-Robot Mapping System

The distributed multi-agent mapping subsystem executes independently on each robot and the GCS. Its primary function is to process sensor data on each robot, share their local mapping data between each robot and the GCS, and to assemble the mapping data into a complete and accurate global map on each robot and the GCS. The system also simultaneously generates accurate localization information for

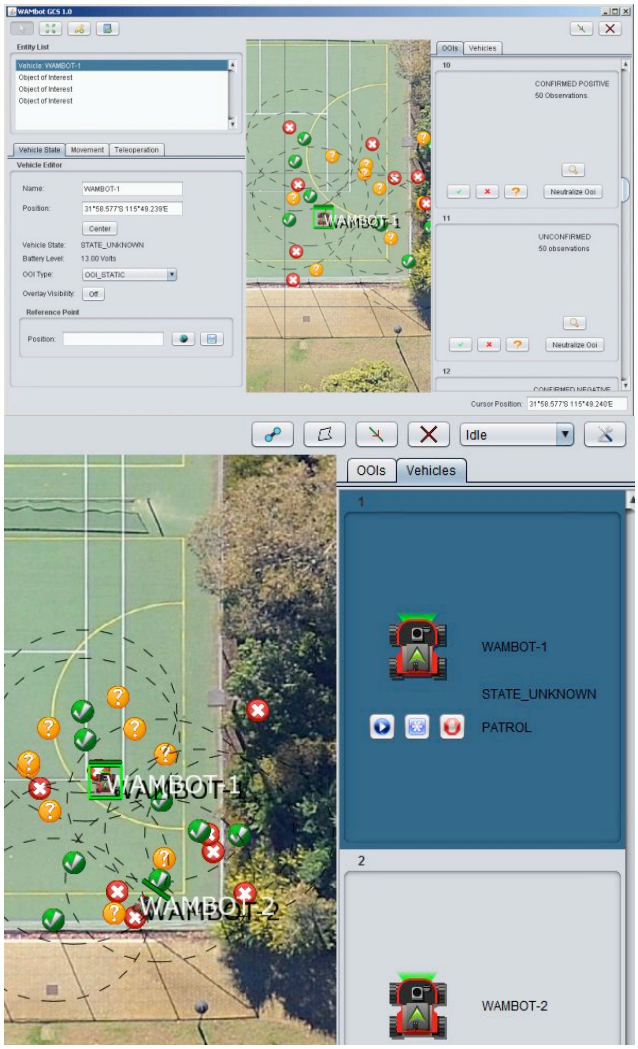

Fig. 3. Ground control station with multi-robot user interface

each robot. At the core of the system is a distributed non-linear optimization engine that efficiently finds the globally-optimal arrangement of the map. The mapping system is described in more detail in [11].

Each robot generates a sequence of small and locally accurate submaps containing 2D spatial information representing their environment. Figure 5 shows sample 30m x 30m submaps from two robots in an outdoor area. Submaps are linked by constraints that describe the relative 2D translation and rotation between them. Each constraint includes probabilistic information (covariance) that forms the "springs" describing the strength of the translation and rotation constraints. These constraints are generated by either relative odometry, global ground-truth measurements (GPS and compass), submap-matching (loop closure), or added by the operator using the GCS. Figure 4 shows the submap and constraint relationships. The compressed submap data and constraints are efficiently broadcast over-the-air to the other robots and the GCS using the distributed data system (DDS).

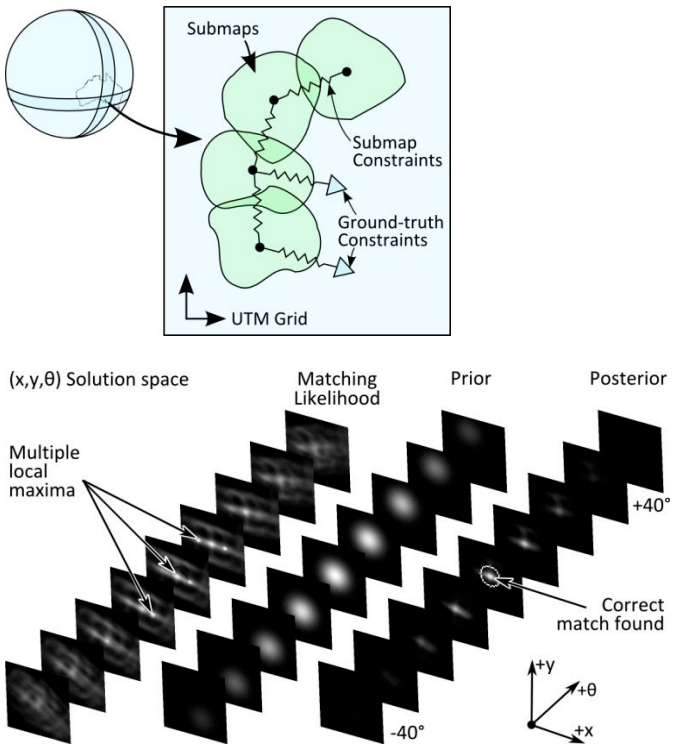

Fig. 4. Sub-maps with constraints and sub-map matching process

Wherever the complete global map is required an incremental global-optimization is performed, followed by an efficient submap composition (fusion) that outputs a globally-optimal map.

The mapping system outputs a 2D occupancy-grid with 0.1m resolution when required. It is an ortho-rectified, globally-aligned image where each 10cm square pixel records the likelihood that it is free-space, unknown, or occupied. Each robot's localization output is accurate to ±0.1m within its local submap. After optimizing with a large number of ground-truth constraints the robot's global pose output is accurate to less than ±0.5m, depending on the constraint graph configuration.

The distributed mapping system can be separated into SLAM front-end and back-end components:

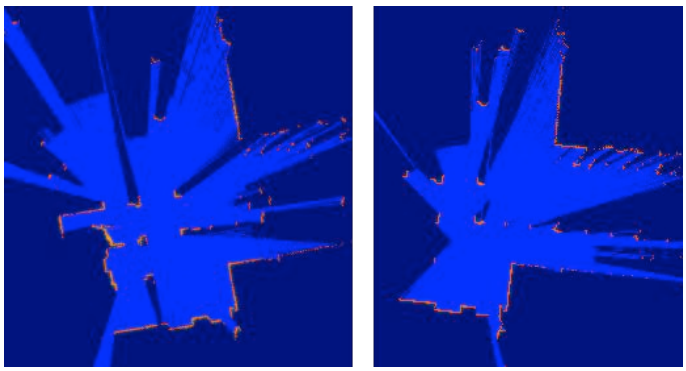

Fig. 5. Example 30m x 30m submaps from two robots outdoors

- Local SLAM: front-end executes in real-time on each robot where it produces submap information. Sensor robots broadcast their submap data over DDS.
- Mapbuilder: forms the SLAM back-end that runs separately on each robot and the GCS. It receives all submap information from the sensor robots and periodically outputs the global map. The Mapbuilder has three sub-components:
  - Optimizer: performs the incremental global map optimization.
  - Builder: composes the submap data and outputs complete global maps.
  - Matcher: searches for spatial matches between submaps and creates additional map constraints; searches for important "loop closures" between all robots.

Instances of Mapbuilder run on each robot and the GCS, with slightly different behaviors depending on the participant type (robot or GCS). The submap optimizer performs an incremental global non-linear optimization on the submap pose estimates based on all of the constraints. The optimization process is decentralized and runs across all robots and at the GCS. The GCS "master" has the ability to "lock down" submaps, minimizing the optimization computation on individual robots.

To join submaps from different robots (multi-robot map fusion), and during loop closures, the Submap Matcher performs a GPU accelerated exhaustive search for spatial matches between submaps. On each robot, the matcher repeatedly attempts to match the robot's current submap against other submaps within its local area. On the GCS the matcher recursively scans the pose-graph looking for spatially close submaps to attempt to match.

IV. Navigation and Planning

The navigation system is responsible for enabling the robots to efficiently move between waypoints without colliding with any obstacles (including other robots) or coming into contact with any hostile objects of interest. Planning is performed on four separate levels: Exploration, High-level, Mid-level and Trajectory planning.

A. Exploration

The exploration planner described in [12] identifies areas to explore and assigns the exploration goals to the appropriate robot. Potential exploration points (frontier points) are identified based on the current robot positions, and the transitions between explored and unexplored terrain. Frontier points are clustered to generate a set of viable frontier regions that could be explored. The potential information gain from each frontier region is estimated through a coarse raycast through each region to determine the characteristics of the unexplored region. Finally the global robot positions, travel distance and frontier region information gain are fused to provide a final destination for each robot.

B. High-Level Planning

The high level planner is responsible for the tactical navigation of the robot fleet. It ensures robots do not approach hostile entities through the use of influence mapping techniques.

The influence map captures a variety of tactical information that can assist the navigation system in determining the optimal movement plan. In the standard influence map approach, hostile objects are given a negative influence and friendly units are given a positive influence. To improve the standard approach, the influence map data is filtered in spatial and time dimensions to provide a moving average of the present tactical situation. In addition to the influence map, a tension and vulnerability map is generated to identify areas of conflict and safe areas respectively. These can be used to move robots into position for neutralization tasks or protect selected robots. This data is used in the high level path planner to generate appropriate paths for the robot.

Each robot then generates a path using a hybrid A*/D* Lite algorithm [6] to solve the goal-directed navigation problem in unknown terrain. D* Lite is a fast replanning method that operates incrementally and efficiently by modifying previous search results locally. It efficiently recalculates a shortest path from its current cell to the goal cell by recalculating only those goal distances that have changed and are relevant for recalculating the shortest path. In the case of a global map update (e.g. a SLAM loop closure event) the path planner performs an initial search of the solution space to generate a partial solution that progresses in the direction of the goal. If the allotted compute time is consumed, the search space is stored and the path planner resumes its search in the next time slot. This enables the robot to continue moving even if finding the optimal path is computationally expensive.

C. Mid-Level Planning

The mid-level path planner performs realtime modifications of the global path to incorporate realtime updates from dynamic obstacles and previously unseen obstacles (see Figure 6). This allows the robot to navigate in a dynamic environment without requiring any user interaction. The WAMbot approach is based on elastic bands [10]. Subjected to artificial forces, the elastic band deforms in real time to a short and smooth path that maintains clearance from all obstacles. Bubbles are represented as point mass particles that are modified according to collision forces between bubbles and the realtime laser scanner data. Higher order integrators and intelligent filtering ensures a high quality path is produced. Consecutive bubbles are constrained to overlap and each bubble constraint is enforced to be wide enough such that the robot can pass through it, ensuring a traversable path.

D. Trajectory Planning

The trajectory planner modifies the vehicle's immediate waypoint and trajectory commands to ensure the vehicle will not collide with an obstacle. The trajectory planner is based on the Dynamic Window Approach [4] that provides goal-directed motion without requiring a map and is capable of moving the mobile base at high velocities while avoiding collisions, even with moving obstacles.

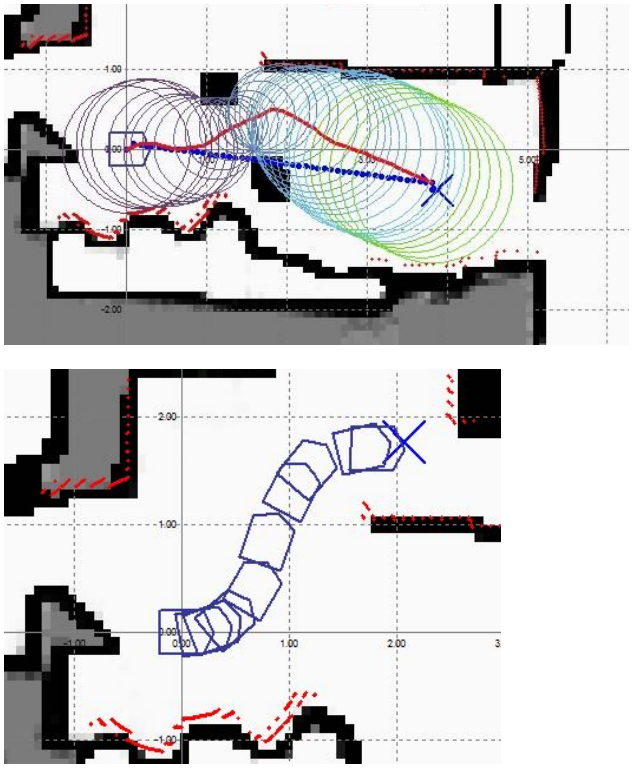

Fig. 6. Mid-level planning using elastic bands and subsequent trajectory planning (images courtesy of Sushil Pangeni, UWA)

V. Object Recognition

Laser scanners are used in combination with digital cameras to assist in the identification and tracking of objects of interest (OOI).

The IBEO Lux laser scanner has a range of 50 meters with four horizontal layers. It enables the WAMbot robots to detect mobile OOI over undulating terrain. The detection of mobile OOI is based on detecting leg-like features and identifying the periodic motion of a walking gait.

In addition, the scanner maintains a database of up to 128 active objects at a given time, assigning them a unique ID and reports on the velocity, size and other characteristics of the objects. The data produced by the LUX is fused with the vision system to generate fused OOI observations.

The identification of static OOIs is based on color and symmetry. As static OOIs are red, they can be detected from a predefined range in the HSV color space. Morphological operations remove noise from the image and the connected components algorithm is used to accurately calculate the centre of mass and size of each OOI using image moments. Potential OOI matches are verified with a color histogram intersection matching method and classified using symmetry lines fitted with the RANSAC algorithm.

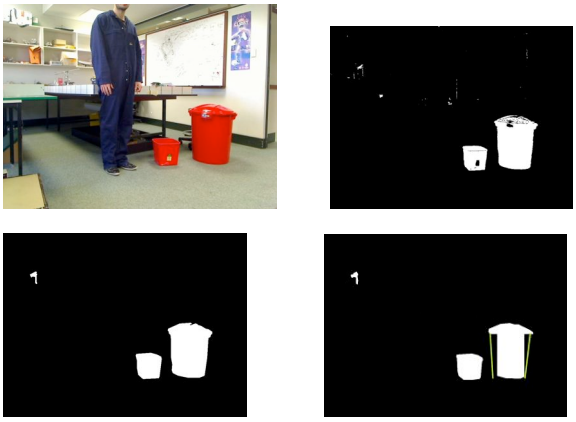

Fig. 7. Image-based object detection: (1) original image, (2) HSV-based thresholding, (3) with morphological operators, (4) with line symmetry (images courtesy of Nicolas Garel, ECU)

Finally, OOI identifications of each robot are combined using a modified expectation maximization algorithm. This fuses and merges all the OOI observations to minimize the observational errors. Mobile OOIs are treated as a special case and have an additional weighted average filter applied, favoring recent observations.

VI. Conclusion

This paper describes the approach Team MAGICian took to meet the MAGIC 2010 requirements and realizing the goal of developing next-generation autonomous ground vehicles. Proven COTS technology, open standards and open source software were leveraged to produce a robust software and hardware solution. Key MAGIC 2010 requirements in sensing, navigation, fusion, mapping, identification and tracking were addressed using state of the art techniques.

Acknowledgments

The authors would like to thank the Western Australian WAMbot team, with members from UWA, ECU, and Thales Australia. Special thanks to A. Attwood, M. Boulton, M. Fazio, B. Frisch, A. Gandossi, N. Garel, M. Hewett, M. Hughes, C.S. Lee, S. Lopes, M. Masek, F. Ophelders, S. Pangeni, Y. Syaid, L. Thin and R. Ucich. We would further like to thank the organizers of MAGIC 2010, RDECOM and DSTO for providing a true robot challenge and also providing team funding. Finally, we would like to acknowledge all our team sponsors, especially Thales Australia.

References

[1] G. Seetharam, A. Lakhotia, E. Blasch, “Unmanned Vehicles Come of Age: The DARPA Grand Challenge”, IEEE Computer, Dec. 2006, pp. 26-29.

[2] C. Urmson, W. Whittaker, “Self Driving Cars and the Urban Challenge”, IEEE Intelligent Systems, March 2008, pp. 66-68.

[3] A. Conradi, L. Foschini, L. Nardelli, “A DDS-compliant infrastructure for fault-tolerant and scalable data dissemination”, The IEEE Symposium on Computers and Communications, June 2010, pp. 489-495.

[4] F. Dieter, W. Burgard, S. Thrun, “Dynamic Window Approach to Collision Avoidance”, IEEE Robotics & Automation magazine, 1997.

[5] Ibeo Automobile Sensor Gmbh, “IBEO Operating Manual”, 2010.

[6] S. Koeni, M. Likhachev, “D* Lite”, Proceedings of the AAAI Conference of Artificial Intelligence (AAAI), 2002.

[7] K. Konolige, K. et al., “Efficient Sparse Pose Adjustment for 2D Mapping”, Intelligent Robots and Systems (IROS 2010), 2010.

[8] E. Olson “Real-time correlative scan matching”, Proceedings of the 2009 IEEE International Conference on Robotics and Automation, 2009, pp. 1233–1239.

[9] F. Lu, E. Milios, “Robot pose estimation in unknown environments by matching 2D range scans”, Journal of Intelligent and Robotic Systems, 18(3), 1997, 249–275.

[10] S. Quinlan, O. Khatib, “Elastic Bands: Connecting Path Planning and Control”, Proceedings of the International Conference on Robotics and Automation, 1993.

[11] R. Reid and T. Bräunl, “Large-scale Multi-robot Mapping in MAGIC 2010,”, Proceedings of the 5th IEEE International Conference on Robotics, Automation and Mechatronics, 2011.

[12] S. Lopes, B. Frisch, A. Boeing, K. Vinsen, T. Bräunl, “Autonomous Exploration of Unknown Terrain for Groups of Mobile Robots”, IEEE Intelligent Vehicles Symposium, Baden-Baden, Germany, June 2011